HTCondor-CE accounting with APEL

Ste Jones
(Liverpool University, GridPP)
GDB - 13 March 2019

Background

- Liverpool T2 uses HTCondor-CE (since January.)
- It works well, but it had no APEL parser to provide accounting.
- We tried HTCondor-CE PIC changes to APEL client parser (single input file). Functionally perfect. But not completely compatible.
- Conflict with CREAM-CE + HTCondor, since it overrides existing HTCondor parser with a new version, and a new file format.

Goals

- Extend APEL parsers to support HTCondor-CE in a way that is compatible with everything else, i.e. no config change required for non-HTCondor-CE sites who update APEL client.
- Directly support HTCondor-CE + HTCondor batch system, but architecture must support other backends, i.e. option to extend further.
- To do this, preserve (to the largest extent) existing data flow/file format conventions, giving minimal code changes (BTW: APEL parser is already well designed in this respect.)
- Release it with UMD as a standard way to link APEL with HTCondor-CE.

Design/implementation

- Retain standard use of two input files; one from CE (blah log), another from batch system (event log).
- Develop data extraction scripts that use HTCondor's powerful (printf-like) formatting language to produce these data files. Hence alternative batch systems are adopted by writing one new data extraction script for the new event log format (just reuse the blah script.)
- Small code change needed (~ 4 lines) for a new, optional scaling factor field in existing HTCondor APEL parser. Transparent to existing CREAM-CE/HTCondor sites. Version 1.8.0-1 of the APEL client parser software contains this change.
- To provide support for heterogeneous clusters, a scheme is used to obtain node scaling factor via some ClassAd. An example scheme is given in the documentation.

Remaining (vaguely related) issue

- New general requirement for all sites using APEL client.
- Until now, APEL client obtains CE benchmark reference via BDII (Glue 1) and puts it in the accounting records to be sent.
- Problems: HTCondor-CE only gives Glue 2; and in any case BDII is soon going away, it is said.
- Solution: ~ 20 line code change to allow admin to hard configure scaling benchmark for CE in the APEL client. No query to BDII. Change awaiting acceptance test/release.
- Note: To “get the show on the road” a workaround is used for the time being - a one-off “static data” SQL insert done by sysadmin/build system. See documentation.

Documentation

End user documentation

https://twiki.cern.ch/twiki/bin/view/LCG/HtCondorCeAccounting#Technical_setup

Scaling factor scheme

https://twiki.cern.ch/twiki/bin/view/LCG/HtCondorCeAccounting#Implement_scaling_factor

Test

https://twiki.cern.ch/twiki/bin/view/LCG/HtCondorCeAccounting#Tests_on_a_HTCondor_CE

Design notes

https://twiki.cern.ch/twiki/bin/view/LCG/HtCondorCeAccountingDesign

Performance in latest test (all Feb)

On the CE:

```
# cd /var/lib/condor/accounting

# ls -rt batch-201902* > /tmp/t

# for f in `cat /tmp/t`; do ~/scripts/workDone.pl < $f;
done | perl -ape'$s+=$F[0]}{$_=$s' ; echo
```

3953875.81 # WHAT WE CLAIM WE SENT FOR FEB.

(workDone.pl audit script listed below)

Performance in latest test (all Feb)

On the EGI Accounting Portal:

Research Infrastructure/T2/UK/NORTHGRID/Feb 2019 → Feb 2019/Sum Wallclock Work/Row Var Submit Host

hepgrid6.ph.liv.ac.uk:9619…

- 3,953,212 # WHAT APEL CLAIMS IT GOT FOR FEB.

Accuracy comparison: ~ 0.017%

Further work

- Find a way to distribute both of the example data extraction scripts. Presently printed in the end user documentation. Maybe bundle them in the UMD APEL client distribution.
- Alternatively, make a simple Puppet module specific to HTCondor-CE APEL client config.
- Etc.

Further ideas

- The idea was to get something working quickly to plug a gap in the functionality. It was easy to modify the APEL client system to provide this new requirement, and it works well.
- But there is a view that a full APEL client system with a mysql database that gives capability for managing multiple CE types at the same site may be over-kill in some simpler situations (not everyone wants to be a DBA.)
- We may need a simpler idea, such as the direct mechanisms for transmitting accounting data from the CE, adopted by (e.g.) VAC.

The workDone.pl audit script

```
# cat workDone.pl
#!/usr/bin/perl
# workDone.pl - independently measure the HS06 hours done in HTCondor-CE batch log file
# Ste Jones, 22 Jan 2019
# Example line:
# batch-20190121-hepgrid6:7017_hepgrid6.ph.liv.ac.uk|prdat128|197|29|3|1548114739|1548114936|19036|58116|
8|0.883|

my $siteNormalisationBenchmark = 10.0;

my $totScaledWallClockSecsWithCores = 0.0;
while (<STDIN>) {
  my $line = $_;
  chomp($line);
  my @fields = split(/\|/,$line);
  my $s    = $fields[2]  * $fields[9] * $fields[10];
  $totScaledWallClockSecsWithCores = $totScaledWallClockSecsWithCores + $s   ;
}
my $hs06Hours = $totScaledWallClockSecsWithCores / 3600.0 * $siteNormalisationBenchmark;

print $hs06Hours,"\n";
```